

**VŠB - Technická univerzita Ostrava**  
**Fakulta elektrotechniky a informatiky**  
**Katedra informatiky**

**Software pro omalovánky**  
**Colouring Book Software**

**2010**

**Jakub Holoubek**

VŠB - Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky  
Katedra informatiky

## Zadání bakalářské práce

Student:

**Jakub Holoubek**

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2612R025 Informatika a výpočetní technika

Téma:

Software pro omalovánky  
Colouring Book Software

Zásady pro vypracování:

Cílem práce je vytvořit aplikaci pro některé mobilní zařízení (např. pro platformu Google Android, Windows Mobile, apod.) umožňující vytvářet předlohy omalovánek a jejich domalování. Požadavky na práci jsou zejména:

1. Využití dotykového ovládání.
2. Schopnost aplikace sdílet předlohy s ostatními uživateli prostřednictvím Internetu.
3. Součástí práce je i jednoduchá webová aplikace umožňující prohlížení předloh, jejich stahování, hodnocení a zobrazování statistik.
4. Možnost testování aplikace prostřednictvím PC emulátorů mobilních zařízení.

Seznam doporučené odborné literatury:


Podle pokynů vedoucího bakalářské práce.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

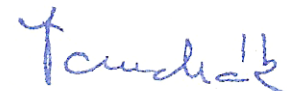
Vedoucí bakalářské práce: **Mgr. Zdeněk Horák**

Datum zadání: 20.11.2009

Datum odevzdání: 07.05.2010

  
\_\_\_\_\_  
doc. Dr.Ing. Eduard Sojka  
vedoucí katedry



  
\_\_\_\_\_  
prof. Ing. Ivo Vondrák, CSc.  
děkan fakulty

## **Prohlášení studenta**

Prohlašuji, že jsem tuto bakalářskou/diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě dne 24. dubna 2010

---

vlastnoruční podpis autora

## **Poděkování**

Děkuji vedoucímu Mgr. Zdeňku Horákovi za cenné rady při řešení problémů. Dále pak Ondřeji Honovi za grafický návrh webové prezentace.

## **Abstrakt**

Bakalářská práce se skládá ze dvou částí. První se zabývá naprogramováním aplikace pro mobilní zařízení s OS Google Android, která bude umožňovat vytváření předloh pro omalovánky a jejich vybarvování. Využívá se platforma Java s Android SDK. Druhá část se zabývá vytvořením webové aplikace, která umožní vytvořené předlohy sdílet mezi více uživateli. Tato aplikace bude vytvořena v jazyce PHP za pomoci knihoven Zend Frameworku. Uživatelé budou mít možnost si jednotlivé předlohy ukládat do svého zařízení, aby je bylo možné vybarvovat také v offline režimu.

## **Klíčová slova**

omalovánky, mobilní zařízení, Google Android, PHP, Zend Framework

## **Abstract**

The bachelor thesis consists of two parts. The first one deals with programming of application for mobile devices running Google Android OS, which will allow the creation of templates for colouring book and their colouring. It uses the Java platform with Android SDK. The second part deals with developing of web site, which will allow share of created templates among users. This site will be developed in PHP using Zend Framework libraries. Users will be able to store each template on their devices, so that they will be able to colour it while being offline.

## **Keywords**

colouring book, mobile device, Google Android, PHP, Zend Framework

## **Použité zkratky**

<b>AJAX</b>	Asynchronous JavaScript and XML
<b>API</b>	Application Programming Interface
<b>HTML</b>	HyperText Markup Language
<b>IDE</b>	Integrated Development Environment
<b>MVC</b>	Model View Controller
<b>OOP</b>	Objektově orientované programování
<b>ORM</b>	Objektově-relační mapování
<b>OS</b>	Operační systém
<b>PDA</b>	Personal Digital Assistant
<b>SDK</b>	Software Development Kit
<b>UI</b>	User Interface
<b>URL</b>	Uniform Resource Locator
<b>WYSIWYG</b>	What You See Is What You Get
<b>XML</b>	eXtensible Markup Language

# Obsah

1. Úvod.....	1
2. Výběr cílové platformy .....	2
2.1. Apple iPhone OS.....	2
2.2. Microsoft Windows Mobile .....	2
2.3. Google Android OS .....	3
2.3.1. Open Handset Alliance .....	3
2.3.2. Open source .....	4
2.3.3. Nejen pro mobilní telefony .....	4
2.3.4. Vývoj aplikací .....	5
3. Webová prezentace: výběr prostředí.....	8
3.1. PHP .....	8
3.1.1. Zend Framework .....	8
3.2. MySQL .....	9
3.3. ORM .....	9
4. Aplikace DroidDrawing.....	10
4.1. Předlohy uložené na internetu.....	11
4.2. Lokální předlohy .....	12
4.2.1. Uložení v interní paměti telefonu.....	12
4.2.2. SQLite databáze .....	12
4.3. Uložení předlohy z Internetu podle ID .....	13
4.4. Nejlépe hodnocené předlohy.....	13
4.5. Uložené vybarvené předlohy .....	14
4.6. Nastavení.....	15
4.7. Tvorba předlohy .....	16
4.7.1. Implementace kreslení .....	17
4.8. Vybarvení předlohy .....	18
4.8.1. Zobrazení informací o předloze a její hodnocení.....	19
4.8.2. Implementace vybarvování .....	19
4.9. Ukázka uložení předlohy na Internet .....	22

5.	Webová prezentace .....	23
5.2.	Omalovánky .....	24
5.3.	Ostatní části webu .....	26
6.	Závěr .....	27
7.	Literatura .....	28
8.	Přílohy .....	29
8.1.	PŘÍLOHA A: Hlavní stránka webové prezentace .....	29



# 1. Úvod

V první části bakalářské práce popíši operační systém Google Android včetně jeho dvou největších konkurentů iPhone OS a Microsoft Windows Mobile. Na OS od Google se zaměřím o něco více a zmíním se o Open Handset Alliance, které je Google zakládající člen a také o tom, jakým způsobem se aplikace pro OS Android vyvíjí.

Další část je zaměřena na výběr prostředí, ve kterém jsem vyvíjel webovou prezentaci pro tento projekt. Proberu platformu PHP, knihovny Zend Framework, ale také použitou MySQL databázi.

Následující kapitola se již věnuje aplikaci DroidDrawing vytvořenou pro OS Google Android. Nejprve nastíním strukturu rozmístění kódu aplikace, následně postupně projdu všechny kroky, které je potřeba provést od vytvoření předlohy, přes její nahrání do webové prezentace až po její vybarvení v mobilním telefonu. V části, která se zabývá vybarvením předloh, navíc vysvětlím princip semínkového algoritmu použitého a proberu některé jeho implementace. Na závěr této kapitoly předvedu, jak nahrát první předlohu na Internet.

Webové prezentaci se bude věnovat další část. Zmíním se zde o svém redakčním systému, jeho struktuře a použitých technologiích. Poté se zaměřím na prezentaci uložených předloh a představím funkce, které systém nabízí. Nakonec uvedu, jakým způsobem se spravují ostatní stránky webové prezentace.

## 2. Výběr cílové platformy

Jak jsem již řekl, mohl jsem si vybrat kterýkoliv existující OS pro mobilní zařízení. Rozhodoval jsem se mezi třemi „kandidáty“ a to mezi OS Apple iPhone<sup>1</sup>, Microsoft Windows Mobile<sup>2</sup> a Google Android<sup>3</sup>.

### 2.1. Apple iPhone OS

Operační systém společnosti Apple Inc. byl poprvé představen v multimediálním zařízení iPod Touch a mobilním telefonu iPhone 9. ledna roku 2007 na konferenci Macworld konané na veletrhu Expo. Prvotní verze 1.0, která obsahovala pouze základní aplikace, byla postupně aktualizována až do současné verze 4.0, která obsahuje řadu nových aplikací a vylepšení. Do této verze byla například přidána jakási obdoba multi-taskingu. Původně firma Apple neuvažovala o tom, že vydá SDK pro vývojáře aplikací třetích stran. Proto bylo ze začátku možné vyvíjet pouze aplikace běžící ve webovém prohlížeči. Toto se ovšem změnilo 6. března 2006, kdy Apple uvolnil první verzi iPhone SDK. Vývoji aplikací třetích stran již téměř nic nebránilo. Z důvodu zachování stability systému je totiž nutné, aby vývojáři dodržovali různá pravidla, která mohou být někdy limitující.

Tento operační systém se používá výhradně na zařízeních firmy Apple. Posledním zařízením využívající tento OS je tablet iPad.

Apple těmito produkty jasně ukázal, jakým směrem by se měla mobilní zařízení ubírat. Vůbec poprvé byl v zařízeních tohoto typu použit „multitouch“ – vícedotykový display. Díky tomu, že Apple vsadil na kvalitní provedení, nádherný design a také - jak bývá u produktů značky Apple zvykem – na jednoduché ovládání, bylo prodáno 6.1 milionů kusů telefonů iPhone za prvních pět čtvrtletí.

### 2.2. Microsoft Windows Mobile

Jedná se o produkt společnosti Microsoft, který je založen na Windows CE. První verze měla název Pocket PC 2000 a byla vydána 19. dubna 2000. Postupem času vycházely verze Windows Mobile 2003, Windows Mobile 5, Windows Mobile 6, až po současnou verzi Windows Phone 7 series. Na rozdíl od produktů Apple, se tento systém objevuje v mobilních zařízeních různých značek. Mezi nejvýznamnější výrobce patří bezpochyby firmy jako HTC, Hewlett-Packard a Samsung.

---

<sup>1</sup> <http://www.apple.com/iphone/>

<sup>2</sup> <http://www.microsoft.com/cze/windowsphone/>

<sup>3</sup> <http://www.android.com/>

## **System se dělí do tří skupin:**

- **Windows Mobile Classic**

- Verze, která je určená pro zařízení typu PDA, tudíž pro zařízení, která neobsahují modul pro telefonování. Vývoj pro tuto skupinu byl vydáním verze 6.5 prakticky ukončen.

- **Windows Mobile Standard**

- Tato verze je již určena pro mobilní telefony, ovšem jen pro ty, které se nedají ovládat pomocí dotykové obrazovky.

- **Windows Mobile Professional**

- Windows Mobile Professional je stejně jako verze Classic určená pro mobilní telefony. Obsahuje však jedno zásadní vylepšení – umožňuje obsluhu zařízení s dotykovou obrazovkou.

## **2.3. Google Android OS**

Pokud bychom si seřadili tyto tři OS podle data uvedení první verze na trh, byl by právě OS od Google tím nejmladším. První prototypy telefonů obsahující Android OS byly představeny 12. února 2008 na události *Mobile World Congress*. Do prodeje se dostal o pár měsíců později. V Americe se začal prodávat 22. října 2008 v telefonu firmy HTC s označením **T-Mobile G1** (T-Mobile přístroj přejmenoval, původní název byl HTC Dream). Prodej v České republice byl zahájen začátkem roku 2009.

OS Android využívá upravenou verzi linuxového jádra. Díky tomu je možné aplikace vyvíjet v programovacím jazyce Java a používat při tom knihovny vyvinuté Googlem. Poslední verze 2.1 byla vydána 12. ledna 2010 a je založena na linuxovém jádře 2.6.29.

### **2.3.1. Open Handset Alliance**

Současně s vydáním OS Android bylo založeno společenství firem, které se budou podílet na vývoji otevřených standardů pro mobilní zařízení. Mezi tyto firmy se řadí mobilní operátoři, softwarové společnosti, výrobci mobilních zařízení a další firmy, které mají co dočinění s mobilními zařízeními. Společně tak chtějí konkurovat firmám, jako jsou Apple, Microsoft, Nokia aj.

### 2.3.2. Open source

Google je znám tím, že většinu svých produktů vydává jako „Open source“. To znamená, že kdokoliv si může zdrojové kódy prohlížet a podle vlastních potřeb upravovat. Výjimkou není ani Android OS. Toho využívají výrobci mobilních zařízení, kteří si tak mohou například upravovat vzhled celého systému. Jako například výrobce mobilního telefonu HTC (viz Obrázek 1).



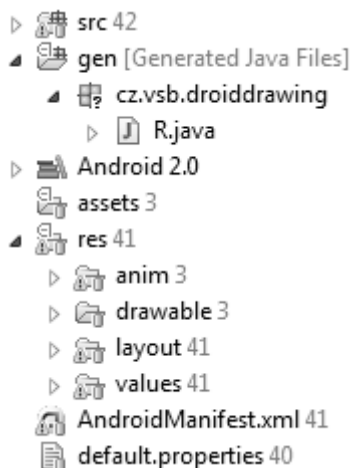
Obrázek 1 - porovnání defaultní úvodní obrazovky a upravená verze na telefonu HTC  
(zdroj: <http://www.mobilewhack.com/tag/sense-ui/>, 20.4.2010)

### 2.3.3. Nejen pro mobilní telefony

I když je OS Android určen převážně pro mobilní telefony, začíná expandovat do ostatních mobilních zařízení, jako jsou tablety a elektronické čtečky knih. V poslední době se také začínají objevovat informace o prvních televizích obsahujících Android OS.

### 2.3.4. Vývoj aplikací

Pro vývoj aplikací byl uvolněn balíček Android SDK. Je možné jej použít na třech nejrozšířenějších OS – Windows, MAC OS, Linux. Obsahuje různé ukázkové aplikace, debugovací nástroje a také emulátor. Oficiálně podporované prostředí pro vývoj



Obrázek 2 - struktura Android aplikace

aplikací je nástroj Eclipse<sup>4</sup> s nainstalovaným pluginem „Android Development Tools“. Existují ovšem pluginy, které umožní vývoj aplikace pro Android také v jiných IDE, např. NetBeans<sup>5</sup>.

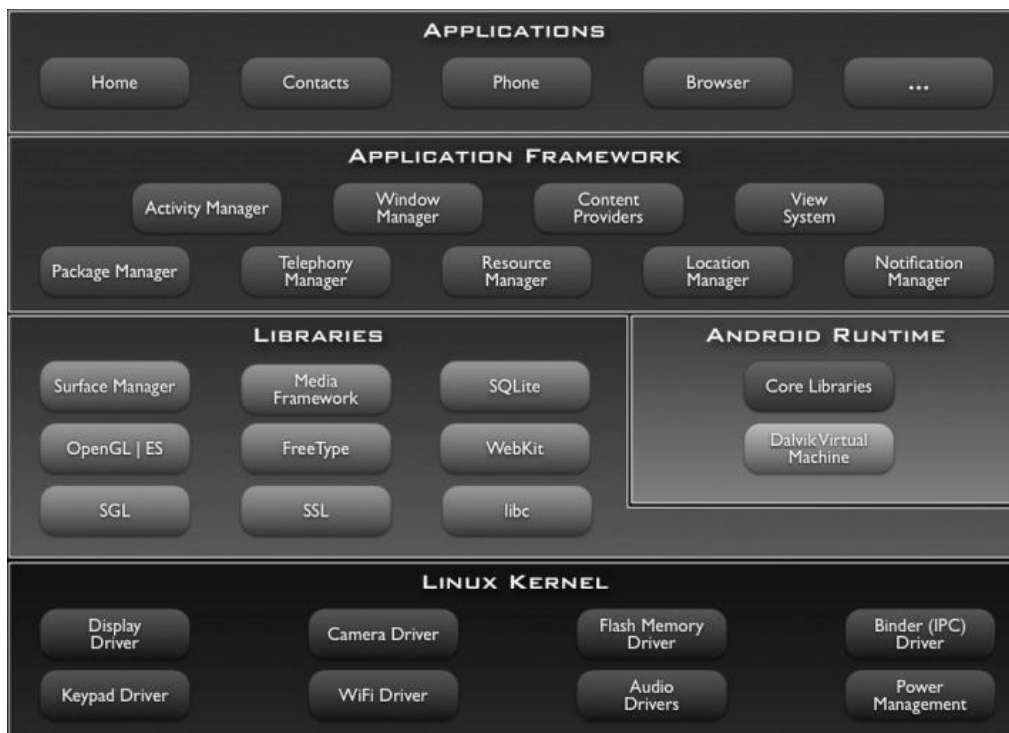
Aplikace se skládá z několika částí. Každé okno je reprezentováno třídou *Activity* a jejími potomky. Základní informace jako název, práva, seznam aktivit aj. jsou uloženy v souboru *AndroidManifest.xml*, který je umístěn v kořenovém adresáři.

Mimo složku *src*, ve které jsou umístěny zdrojové soubory, je zde složka *gen*, která obsahuje automaticky generovanou třídu *R.java*. Tato třída obsahuje odkazy na objekty uložené ve složce *res*, jejíž obsah popíši později. Dále jsou zde Android SDK knihovny. Do složky *assets* se ukládají soubory, které se čtou po jednotlivých bytech, jako je audio a video. Adresář *res* obsahuje převážně XML soubory, které definují rozvržení komponent na obrazovce, animace a také nastavení textových řetězců, které se v aplikaci používají. Posledním souborem je automaticky generovaný *default.properties*, ve kterém je např. nastavena verze SDK. Obrázek 2 ukazuje celkovou strukturu aplikace pro Android OS.

---

<sup>4</sup> <http://www.eclipse.org/>

<sup>5</sup> <http://netbeans.org/>



Obrázek 3 - architektura  
(zdroj: <http://developer.android.com/guide/basics/what-is-android.html>, 20.4.2010)

Android obsahuje několik základních knihoven napsaných v jazycích C a C++, ke kterým smí aplikace přistupovat prostřednictvím aplikačního frameworku. Jsou zde například knihovny renderovacího jádra webového prohlížeče „*WebKit*“, které se používá také u desktopových webových prohlížečů, jako např. Google Chrome<sup>6</sup> anebo Safari<sup>7</sup>. Podpora pro mediální soubory je v těchto knihovnách také obsažena.

Android SDK obsahuje několik základních aplikací jako je E-mailový klient, aplikace pro správu SMS zpráv aj. Vývojáři mají přístup k naprosto stejnému API, které tyto základní aplikace využívají.

<sup>6</sup> <http://www.google.com/chrome>

<sup>7</sup> <http://www.apple.com/safari>

### 2.3.4.1. Třída Activity

Jak už jsem se zmínil, každé okno aplikace je reprezentováno potomky třídy „*Activity*“. Třída dědicí z „*Activity*“ se tedy stará o vytvoření okna, které můžeme naplnit vlastním uživatelským rozhraním většinou umístěným ve složce „*res/layout*“. Všechny aktivity musí mít záznam v souboru *AndroidManifest.xml* obsahující atribut „*android:name*“, který odkazuje na třídu implementující tuto aktivitu. Většina aktivit pokrývá celou obrazovku zařízení, je však možné toto implicitní chování změnit. Můžeme tak například vytvořit aktivity vypadající jako dialogové okno, pod kterým bude vidět předchozí aktivita. Všechny spuštěné aktivity Android ukládá do datové struktury založené na principu zásobníku. Aktivita na popředí je v této struktuře umístěna úplně nahoře. Stisknutím tlačítka ZPĚT na telefonu se aktivita na popředí ukončí a zobrazí se předchozí aktivita. Stisknutím tlačítka HOME se aktivita na popředí přesune na druhé místo zásobníku a na první se přesune aktivita reprezentující úvodní obrazovku.

### 2.3.4.2. Tvorba uživatelského rozhraní

Pro tvorbu uživatelského rozhraní obsahuje Android SDK několik komponent. Základním prvkem je třída s názvem „*ViewGroup*“. Komponenty dědicí z této třídy mohou obsahovat jiné komponenty.

**Existuje šest základních komponent dědicích z ViewGroup:**

- *LinearLayout*
- *RelativeLayout*
- *TableLayout*
- *GridView*
- *ListView*
- *TabLayout*

Komponenty, které nemohou obsahovat jiné komponenty, dědí ze třídy „*View*“ a říká se jim „*widgety*“. Existují widgety, které mohou být použity pouze v určité *ViewGroup*, jako např. *<TableRow>* umístěný v *TableLayout*. Vývojář má možnost si vzhled každé komponenty upravovat jak uzná za vhodné anebo si může vytvořit svou vlastní komponentu.

## 3. Webová prezentace: výběr prostředí

Zadání bakalářské práce bylo rozděleno do dvou částí. První část se týkala vytvoření vlastní aplikace pro Android OS a druhá obsahovala vytvoření webové prezentace, ve které bude katalog zhotovených předloh. K vytvoření stránek jsem použil jazyk PHP se Zend Frameworkem. Jako databázi jsem zvolil MySQL. K propojení databáze a PHP jsem použil vlastní ORM systém.

### 3.1. PHP<sup>8</sup>

Jedná se o široce používaný skriptovací jazyk primárně určený pro tvorbu webových prezentací. Vyvinul jej dánský programátor Rasmus Lerdorf. První verzi PHP 1.0 vydal 8. června 1995. V roce 1998 se k Lerdorfovi přidali dva izraelští vývojáři Zeev Suraski a Andi Gutmans. V červnu roku 2004 byla vydána verze PHP 5 s vylepšenou podporou OOP. Poslední verze je PHP 5.3.2.

Velkou výhodou PHP je to, že je multiplatformní. V kombinaci s linuxovými servery, webovým serverem Apache a MySQL databází jsou proto provozní náklady prakticky nulové. Díky tomu existuje mnoho hostingových serverů, které PHP provozují za příznivou cenu.

#### 3.1.1. Zend Framework<sup>9</sup>

Zend Framework je aplikační framework vyvinutý přímo tvůrci jazyka PHP. Jeho první verze byla uvolněna 1. července 2007. Aplikace vytvořené pomocí Zend Frameworku využívají návrhový vzor MVC, což rozděluje zdrojový kód do tří částí:

- **MODEL** správa dat, se kterými aplikace pracuje
- **VIEW** použití dat, nastavené modelem a jejich zobrazení uživateli
- **CONTROLLER** reakce na události uživatele, změny v modelu a ve view

Každá třída controlleru obsahuje tzv. „akce“, což jsou metody s názvem ve formátu `<nazevAkce>Action()`. Konkrétní controller a akce, která se má použít, jsou určeny podle URL adresy. Implicitně je nastaveno, že první část adresy za doménou určuje controller, druhá část akci a každá následující dvojice určuje název parametru a jeho hodnotu. URL adresa „clanky/zobraz/id/3“ by proto provedla akci „zobraz“ v controlleru „ClankyController“ a vytvořila by parametr id s hodnotou 3.

---

<sup>8</sup> <http://cz.php.net/>

<sup>9</sup> <http://framework.zend.com/>



Zend Framework dále obsahuje knihovny pro snadné připojení k několika nejvýznamnějším databázím, jako jsou Oracle, MySQL, Microsoft SQL Server aj.

Současná verze nese označení 1.10 a je k dispozici včetně javascriptové knihovny DOJO<sup>10</sup> pro tvorbu aplikací, splňující zásady „WEB 2.0“.

### 3.2. MySQL<sup>11</sup>

Původně produkt firmy MySQL AB. V roce 2008 jej získal Sun Microsystems, který 27. ledna 2010 koupila firma Oracle Corporation. Jedná se o relační databázový server, který existuje jak v placené, tak i ve volně dostupné verzi. Ve spojení s webovými stránkami se jedná bezesporu o nejpoužívanější databázový server. Od verze 5.0, která byla vypuštěna v říjnu 2005, je rozšířena o možnost vytváření uložených procedur, triggerů, pohledů aj.

Po pohlčení společnosti Sun Microsystems Oraclem se spekulovalo o tom, co se stane s jejími produkty. Představitelé Oracle však všechny ujistili v tom, že přestože je MySQL konkurencí databázového serveru od Oracle, budou se snažit jej zachovat a dokonce se budou podílet na vylepšení tohoto oblíbeného serveru.

### 3.3. ORM

Objektově-relační mapování je technika, která převádí data relační databáze na objekty používané v objektově-orientovaných programovacích jazycích. Díky této technice se vytvoří jakási kopie databázové struktury v podobě objektů. Implementací ORM systémů existuje celá řada. Mezi nejpoužívanější patří Hibernate<sup>12</sup> pro jazyk Java anebo LINQ to SQL<sup>13</sup> pro jazyky využívající .NET Framework.

---

<sup>10</sup> <http://www.dojotoolkit.org/>

<sup>11</sup> <http://www.mysql.com/>

<sup>12</sup> <http://www.hibernate.org/>

<sup>13</sup> <http://msdn.microsoft.com/en-us/library/bb425822.aspx>

## 4. Aplikace DroidDrawing



Když jsem začínal tuto aplikaci vyvíjet, byla aktuální verze OS Android 2.0, proto je kompatibilní s touto verzí OS. Zdrojové kódy jsou rozmístěny v několika balících.

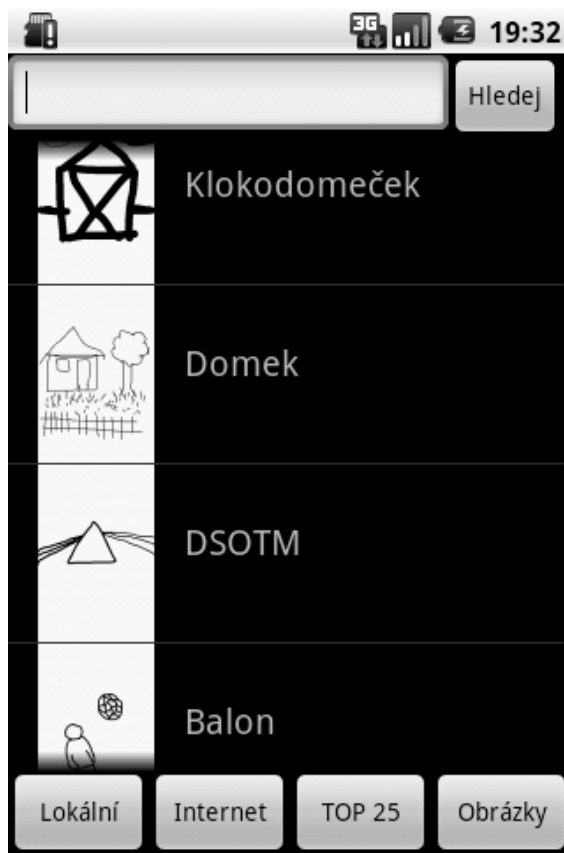
- **cz.vsb.painting**
  - hlavní aktivity a ostatní nezařazené třídy
- **cz.vsb.painting.adapters**
  - adaptory sloužící pro připojení k databázi aj.
- **cz.vsb.painting.dialogs**
  - třídy reprezentující dialogová okna
- **cz.vsb.painting.tasks**
  - soubory např. pro obsluhu nahrávání předloh na Internet
- **cz.vsb.painting.views**
  - modifikované/vytvořené UI komponenty

```
▷ cz.vsb.droiddrawing 41
▷ cz.vsb.droiddrawing.adapters 41
▷ cz.vsb.droiddrawing.dialogs 41
▷ cz.vsb.droiddrawing.tasks 41
▷ cz.vsb.droiddrawing.views 41
```

Obrázek 4 - struktura aplikace DroidDrawing

Aplikace je rozdělena do těchto částí:

- Poslední předlohy uložené na internetu
- Lokální předlohy
- Uložení předlohy z internetu podle ID
- Nejlépe hodnocené předlohy
- Uložené vybarvené předlohy
- Nastavení
- Tvorba předlohy
- Vybarvení předlohy
- Zobrazení informací o předloze



Obrázek 5 - úvodní obrazovka

Některé z těchto částí jsou přístupny pouze uživatelům, kteří jsou registrováni ve webové prezentaci. Funkce jednotlivých sekcí nyní podrobněji popíši.

#### 4.1. Předlohy uložené na internetu

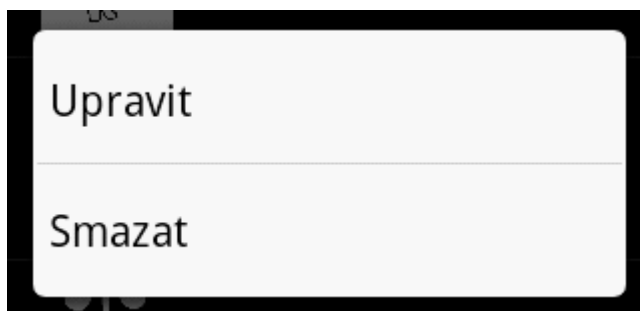
Tato sekce se zobrazí ihned po spuštění aplikace. Již z nadpisu je jasné, že se zde zobrazují ty předlohy, které jakýkoliv uživatel nechal sdílet s ostatními. Načítá se zde pouze posledních 25 předloh. Jelikož se veškerá data stahují z Internetu a načtení 25-ti obrázků může být někdy příliš zdlouhavé, implementoval jsem proto tuto část tak, že se nejprve stáhne veškerý text a miniatury předloh se začnou stahovat teprve tehdy, pokud jsou vidět na obrazovce.

Veškerá textová data stažená z Internetu jsou webovou prezentací převáděna do XML souboru, obsah tohoto souboru si mobilní aplikace stáhne a následně parsuje.

Díky textovému poli umístěnému nad seznam předloh může uživatel vyhledávat předlohy podle názvu. Všechny položky ze seznamu je možné uložit do lokálních předloh. Stačí dlouze kliknout na zvolenou předlohu a objeví se nám kontextové menu s možností „Uložit do lokálních“.

## 4.2. Lokální předlohy

Jedná se o ty předlohy, které buďto vytvořil uživatel sám anebo si je uložil z Internetu. Tyto předlohy lze poté používat i v tom případě, že není dostupné internetové připojení. Předlohy, které vytvořil sám uživatel, mohou být dále upravovány. U předloh stažených z Internetu je umožněno pouze jejich vybarvení a smazání. *Obrázek 6* ukazuje kontextové menu, které se zobrazí u uživatelem vytvořené předlohy.



Obrázek 6 - kontextové menu

K uchovávání těchto dat v telefonu používám:

- Interní paměť telefonu
- SQLite<sup>14</sup> databáze

### 4.2.1. Uložení v interní paměti telefonu

Tímto způsobem se do datové struktury mobilní paměti ukládají obrázky ve formátu PNG. Podle jejich názvů se poté jednotlivé soubory spojují se záznamem v databázi.

### 4.2.2. SQLite databáze

Data lze v OS Android ukládat několika způsoby:

- Uložení do paměti telefonu
- Využití tzv. „Shared Preferences“ – pro základní datové typy
- Použití SQLite databáze

---

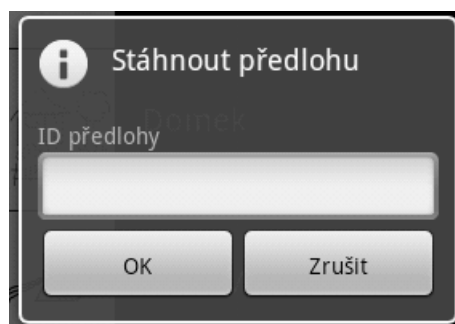
<sup>14</sup> <http://www.sqlite.org/>

SQLite je relační databáze, která nepracuje na principu „klient-server“, ale je připojena v podobě malé knihovny ke každé aplikaci. Samotná databáze se ukládá do souboru s příponou .dbm uloženého v paměti telefonu.

SQLite databázi využívám k zaznamenání detailnějších informací o každé předloze uložené v telefonu.

### 4.3. Uložení předlohy z Internetu podle ID

Uživatel si smí vybrat předlohu ve webové prezentaci a pomocí jedinečného ID čísla si ji uložit do lokálních předloh. Stačí vybrat položku „*Stáhnout předlohu*“ z menu. Objeví se dialogové okno (viz *Obrázek 7*) obsahující textové pole pro zadání ID předlohy.



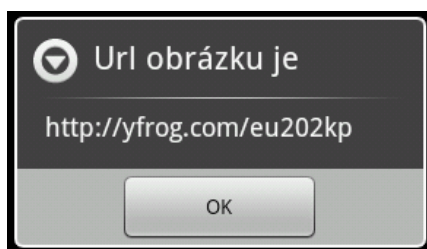
Obrázek 7 - stažení předlohy z internetu

### 4.4. Nejlépe hodnocené předlohy

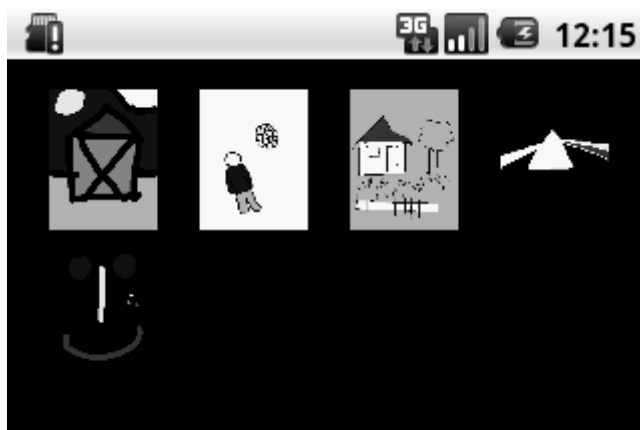
Výpis je téměř totožný s položkou „Předlohy uložené na internetu“, neobsahuje ovšem možnost vyhledávání. Namísto seřazení položek podle data nahrání na web se zde předlohy řadí podle průměrného hodnocení, které jim udělili uživatelé.

## 4.5. Uložené vybarvené předlohy

Všechny vybarvené předlohy je možné uložit do paměti telefonu. Výpis je realizován pomocí miniatur obrázků (viz *Obrázek 9*). Uložené obrázky lze následně upravovat, mazat a také ukládat na populární obrázkový hosting „ImageShack.US“. Po nahrání obrázku na server se zobrazí dialogové okno s URL adresou nahraného obrázku.



Obrázek 8 – uložení obrázku na ImageShack.us



Obrázek 9 – výpis vybarvených předloh

## 4.6. Nastavení

Pro ukládání nastavení aplikací se v OS Android používají již zmíněné „Shared Preferences“. Tato nastavení se ukládají formou XML souboru v domovské složce aplikace. Vytvoření a správa toho souboru je čistě v režii operačního systému.

V nastavení aplikace se mohou uživatelé buďto přihlásit, a pokud nejsou zaregistrováni, tak také provést samotnou registraci. Po otevření nabídky „Registrovat“ se zobrazí formulář (viz *Obrázek 10*). Jakmile se uživatel úspěšně zaregistruje, může se přihlásit pomocí dialogového okna (viz *Obrázek 11*).



Obrázek 10 - registrace uživatele



Obrázek 11 - přihlášení

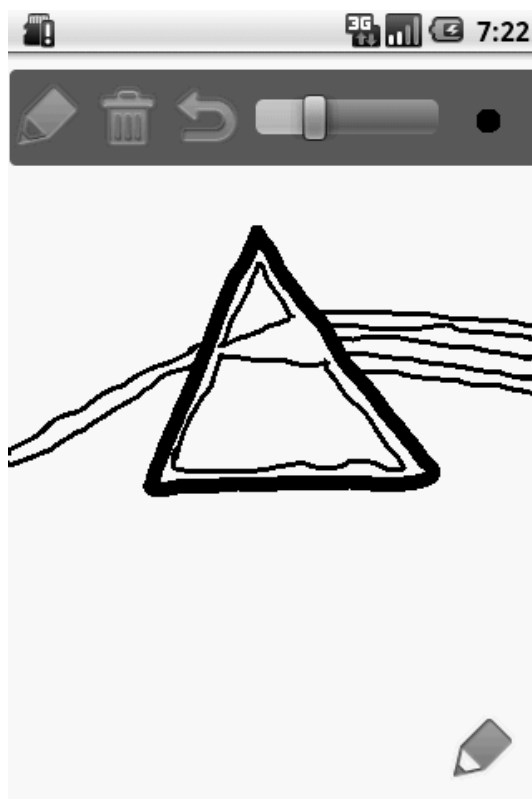
Po úspěšném přihlášení se login aktuálně přihlášeného uživatele zobrazí v druhém řádku tlačítka „Odhlásit“ (viz *Obrázek 12*).



Obrázek 12 - přihlášený uživatel

## 4.7. Tvorba předlohy

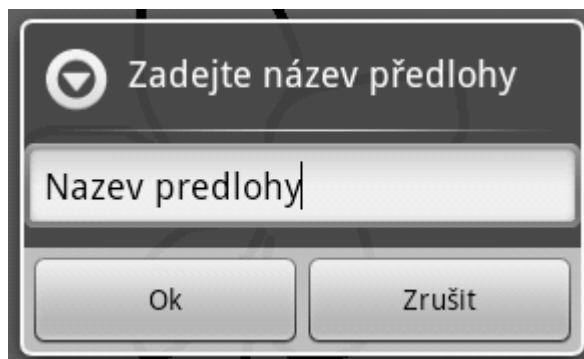
Tvorba předlohy je jednou z hlavních částí celé aplikace. Pro otevření kreslicího plátna stačí vybrat položku „Vytvořit nový“ z menu aplikace. Snažil jsem se využít maximální možnou plochu, proto zde nenajdeme žádné nadbytečné ovládací prvky. Ihned po otevření se zobrazí bílá plocha s ikonou tužky, která je umístěna vpravo dole. Bez nutnosti nastavovat další parametry můžeme začít vytvářet novou předlohu. Po kliknutí na ikonu tužky se nám přece jen zobrazí menu, obsahující základní prvky pro přizpůsobení kreslené čáry. *Obrázek 13* nám ukazuje celou obrazovku tvorby předlohy včetně zobrazeného menu. V menu máme na výběr z několika možností. Po kliknutí na symbol tužky můžeme na plochu kreslit černé čáry. V případě, že se nám něco nepovedlo, můžeme kliknout na ikonu koše, čímž se změní barva na bílou a máme možnost nakreslené čáry vymazat. Pokud nám nebude stačit mazání ručně, můžeme využít třetí prvek, kterým je tlačítko zpět, díky kterému se můžeme vrátit o několik kroků nazpátek. Aplikace si pamatuje poslední 4 kroky. Dalším prvkem v menu je posuvník, kterým nastavujeme šířku čáry v rozmezí od dvou do padesáti bodů. K tomu, abychom ihned viděli nastavenou šířku, slouží poslední prvek. Jedná se o černou tečku, která mění svůj průměr, jakmile se změní hodnota posuvníku.



Obrázek 13 - tvorba předlohy



Jakmile jsme s vytvářením předlohy hotovi, přistoupíme k jejímu uložení. Po stisknutí tlačítka menu se zobrazí dvě nabídky a to „Uložit do lokálních“ a „Uložit na web“. Pokud nechceme námi vytvořenou předlohu sdílet se všemi uživateli, zvolíme položku „Uložit do lokálních“. Zobrazí se dialogové okno pro zadání názvu předlohy (viz *Obrázek 14*). Později je samozřejmě možné, že si to uživatel rozmyslí a bude se chtít o svůj výtvar podělit s ostatními uživateli. V tom případě zvolí položku „Uložit na web“. Jestliže není přihlášen, zobrazí se nejprve dialogové okno pro přihlášení a poté výzva k zadání názvu předlohy. Následně proběhne odeslání dat do webové prezentace.



Obrázek 14 - zadání názvu předlohy

#### 4.7.1.Implementace kreslení

Zpočátku jsem kreslení implementoval tak, že jsem si nakreslené čáry ukládal ve formě objektů do kolekce. Tento způsob jsem měl vyzkoušen na desktopové aplikaci, kde vše fungovalo bez problémů. Po chvíli testování jsem však zjistil, že s rostoucím počtem čar rapidně klesá výkonnost celé aplikace, proto jsem musel hledat jiné řešení. Nakonec jsem vše vyřešil úplně jinak. Vzhled každé komponenty je definován objektem typu *Canvas*, který obsahuje bitmapu. Změny na této bitmapě se projeví ve vzhledu komponenty. Vytvořil jsem proto novou komponentu, která reaguje na pohyb prstu. Při položení prstu na tuto komponentu se v paměti vytvoří objekt typu *Path*, který bude uchovávat informace o kreslené čáře. Pohybem prstu se budou do tohoto objektu přidávat body, mezi kterými se nakreslí čára. Pro názornost jsem tento proces popsal pseudokódem (viz *Pseudokód 1*).

**Kresli(akce, x, y):**

Pokud se jedná o akci stisknutí tak:

Vyčisti cestu čáry Path.

Nastav poslední bod na bod se souřadnicemi x, y.

Pokud se jedná o akci pohybu tak:

Přidej do Path trasu z posledního bodu do bodu se souřadnicemi x, y.

Nastav poslední bod na bod se souřadnicemi x, y.

Pokud se jedná o akci uvolnění tak:

Přidej do Path trasu z posledního bodu do bodu se souřadnicemi x, y.

Nakresli do bitmapy cestu z Path.

Vyčisti Path.

KONEC.

Pseudokód 1 - kreslení čar

## 4.8. Vybarvení předlohy

Okno, ve kterém se vybarvují předlohy, lze otevřít ze všech čtyř částí dostupných na úvodní obrazovce, stačí kliknout na libovolnou předlohu. Toto okno je téměř totožné s tím, ve kterém se předlohy vytváří. Rozdíl je v tom, že tlačítko zpět je umístěno v levém spodním rohu a místo tužky je v pravém spodním rohu umístěn čtverec, který otevírá dialogové okno pro výběr barvy (viz Obrázek 15). Po zvolení barvy se změní barva také tohoto čtverce.



Obrázek 15 - vybarvení předlohy

Vybarvené předlohy můžeme uložit, když vybereme položku „Uložit“ z menu. Jestliže je předloha stažena z Internetu, zobrazí se nám v menu také položka „Info“.

#### 4.8.1. Zobrazení informací o předloze a její hodnocení

Kliknutím na tuto položku se otevře dialogové okno (viz *Obrázek 16*), ve kterém najdeme podrobnější informace o předloze. Je tady zobrazen název předlohy, její tvůrce, datum nahrání na server, průměrné hodnocení a také hodnocení uživatele.

Hodnocení se provádí tak, že uživatel nastaví počet hvězd v rozmezí od 1 do 5. Jakmile předlohu ohodnotí, odešle se informace o hodnocení na server. Uživatel může ohodnotit každou předlohu pouze jedenkrát.



Obrázek 16 - info o předloze

#### 4.8.2. Implementace vybarvování

Pro řešení problému vybarvení jednobarevné plochy jinou barvou se nejčastěji používá „Semínkový algoritmus“ v angličtině se mu říká „Seed fill“ nebo také „Flood Fill“. Tento algoritmus prochází bod po bodu a kontroluje, zda je barva bodu totožná s tou, jakou má bod, od kterého se začalo vyhledávat. Pokud se barvy shodují, nalezený bod přebarví novou barvou. Na stejném principu funguje např. známá hra „Hledání min“.

Existuje několik typů tohoto algoritmu, každý se dá napsat buďto pomocí rekurze anebo pomocí zásobníku. Použití rekurze se ovšem příliš nedoporučuje, při vybarvování větších ploch totiž může dojít k přetečení zásobníku paměti. Některé z těchto implementací semínkového algoritmu podrobněji popíši.

#### 4.8.2.1. 4cestný semínkový algoritmus

Z pohledu implementace se jedná o nejjednodušší algoritmus, je ovšem také nejpomalejší. Jako parametry funkce implementující 4cestný semínkový algoritmus se používají: *souřadnice bodu*, *současná barva* a *nová barva*. Tato funkce je pak pokaždé zavolána čtyřikrát (prochází se 4 sousední body), dokud je barva bodu stejná jako parametr „*současná barva*“, a zároveň jiná než nová barva. Ukázka - viz *Pseudokód 2*.

**FloodFill4(x, y, NováBarva, SoučasnáBarva):**

Pokud je barva bodu (x,y) jiná než SoučasnáBarva:

KONEC.

Nastav barvu bodu (x,y) na NováBarva

Proveď FloodFill4 s bodem nalevo od bodu (x,y).

Proveď FloodFill4 s bodem napravo od bodu (x,y).

Proveď FloodFill4 s bodem, který je nad bodem (x,y).

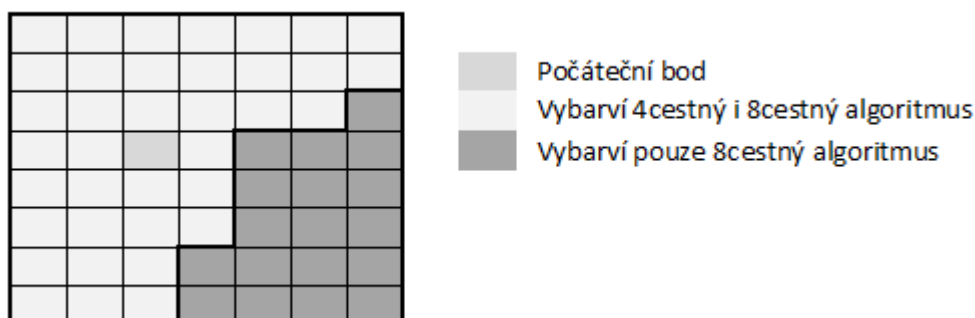
Proveď FloodFill4 s bodem, který je pod bodem (x,y).

KONEC.

Pseudokód 2 – 4cestný semínkový algoritmus

#### 4.8.2.2. 8cestný semínkový algoritmus

Implementace je velmi podobná předchozímu algoritmu. Jediný rozdíl je v tom, že se prochází všechny sousední body, tedy i ty, které mají společný pouze roh. Výsledkem je, že se tento algoritmus zpracuje i ty body, ke kterým by se 4cestný algoritmus nedostal (viz *Obrázek 17*). Nevýhodou obou algoritmů je to, že každý bod je kontrolován vícekrát, což má nepříznivý vliv na výkonnost.



Obrázek 17 - porovnání algoritmů

### 4.8.2.3. Scanline algoritmus

Jako poslední popíší algoritmus „*Scanline*“, který jsem nakonec použil ve své aplikaci. Z těchto tří algoritmů je nejrychlejší, jeho implementace je ovšem tou nejsložitější. Počáteční bod přidá do fronty, poté mu nastaví novou barvu. Následně zkontroluje body napravo a nalevo, pokud mají cílovou barvu (původní barva počátečního bodu), tak je přidá do fronty také. Pokračuje dále tím, že sníží hodnotu souřadnice Y současného bodu o jedna. To opakuje, dokud nenalezne bod, který nemá cílovou barvu anebo dokud nenarazí na okraj obrazovky. Výhodou tohoto algoritmu je, že každý bod zkontroluje pouze jednou, proto je několikanásobně rychlejší, než předchozí dva příklady.

Původně jsem neprovedl převedení bitmapy do pole čísel, což mělo za následek to, že vybarvení celé plochy obrazovky, která má rozlišení 320x480px, trvalo déle než 10 vteřin a někdy aplikace dokonce přestala pracovat úplně. Díky tomu, že jsem poté použil převod do pole, se doba vybarvení celé obrazovky zkrátila na cca 3.5 vteřiny. Nastínění funkce algoritmu - viz *Pseudokód 3*.

#### **Vybarvi(PocatecniBod, Bitmapa, CílováBarva, NováBarva):**

Nastav Q na prázdnou frontu.

Převed' Bitmapu do pole čísel C, reprezentujících barvu jednotlivých pixelů.

Přidej počáteční bod do fronty Q.

Dokud není Q prázdná tak:

    Odeber první prvek z Q a nastav jím proměnnou P.

    Nastav proměnnou Y na y-souřadnici bodu P.

    Dokud je Y větší než 0 a barva pixelu Bitmapy na souřadnici (P.X, Y) je CílováBarva tak:

        Zmenši Y o jedna.

    Dokud má bod (P.X, Y) CílovouBarvu tak:

        Změň barvu pixelu v poli C o souřadnicích (P.X, Y) na NovouBarvu.

        Pokud má bod nalevo od bodu (P.X,Y) CílovouBarvu, tak jej přidej do Q.

        Pokud má bod napravo od bodu (P.X,Y) CílovouBarvu, tak jej přidej do Q.

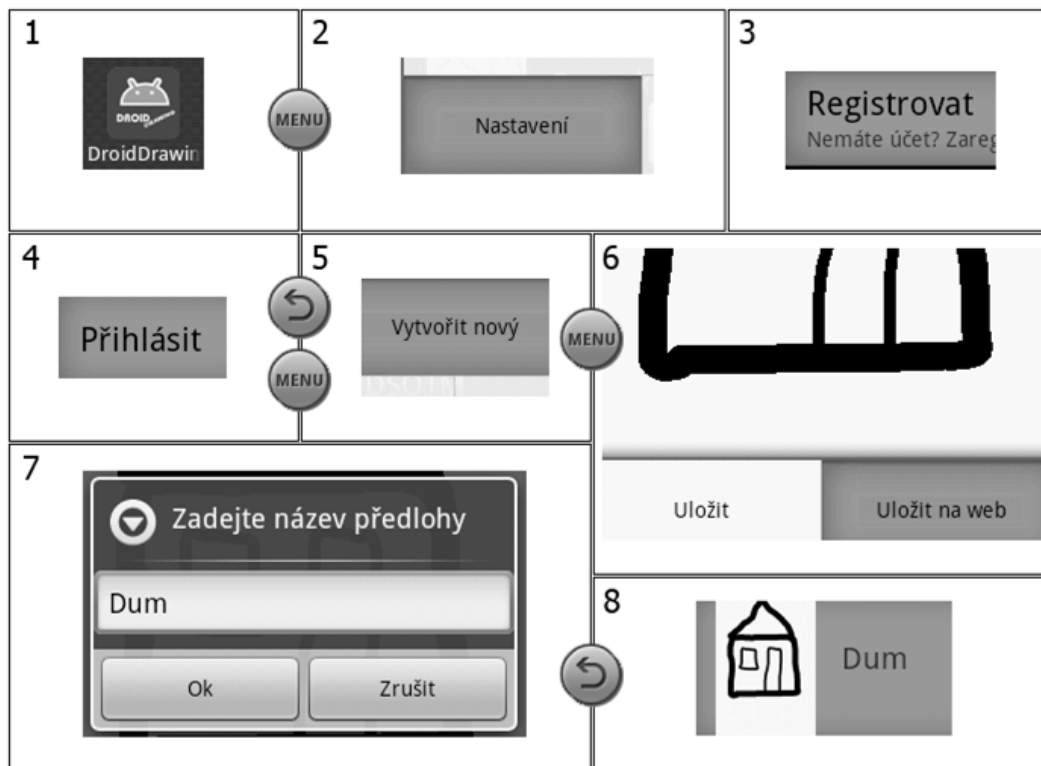
        Zvětši Y o jedna.

    Převed' pole C do bitmapy a vykresli ji.

KONEC.






**Pseudokód 3 - vybarvení předlohy**

## 4.9. Ukázka uložení předlohy na Internet



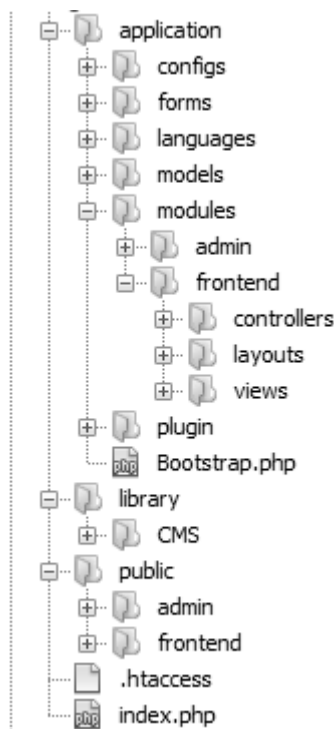
Obrázek 18 - návod: Uložení předlohy na Internet

### Popis kroků:

1. Spuštění aplikace v menu telefonu
2. Po stisknutí tlačítka  se objeví seznam položek, vybereme „Nastavení“
3. Pokud nejsme zaregistrováni, klikneme na tlačítko „Registrovat“
4. Po úspěšné registraci se přihlásíme kliknutím na tlačítko „Přihlásit“
5. Stisknutím tlačítka  se vrátíme na úvodní obrazovku, zde stiskneme tlačítko  a vybereme nabídku „Vytvořit nový“
6. Jakmile máme předlohu vytvořenou, stiskneme  a vybereme „Uložit na web“
7. Objeví se dialogové okno, ve kterém vyplníme název předlohy a potvrdíme jej stisknutím tlačítka „Ok“
8. Jakmile se předloha odešle, stiskneme tlačítko  . Načte se seznam předloh uložených na Internetu obsahující také námi vytvořenou předlohu. Po kliknutí ji můžeme začít ihned vybarvovat.

## 5. Webová prezentace

K vytvoření webové prezentace jsem použil svůj jednoduchý PHP redakční systém využívající Zend Framework, do kterého jsem přidal modul pro zobrazování předloh a informací o nich. Aplikace je rozdělena do poměrně rozsáhlé struktury (viz *Obrázek 19*). Hlavním souborem je bezpochyby „*index.php*“, na který se přesměrovávají veškeré odkazy.



Obrázek 19 - struktura webové aplikace

V tomto souboru se nastavují cesty k některým hlavním adresářům a také globální proměnné. Ve složce „*application*“ umístěny soubory obsahující jak aplikační logiku, tak HTML šablony. Důležitým souborem je zde „*Bootstrap.php*“, ve kterém se inicializuje připojení k databázi a směrování URL adres na konkrétní controller. Složka „*configs*“ obsahuje konfigurační soubory s příponou *ini*. Jsou zde například nastaveny přístupové údaje k databázi. V další složce s názvem „*forms*“ se nachází soubory používané ke generování formulářů. Jedná se o obyčejné třídy, které jsou potomky třídy „*Zend\_Form*“, ta například poskytuje rozhraní pro snadnou validaci formulářů. Do adresáře „*languages*“ se umísťují překlady textových řetězců do různých jazyků. V mém případě je zde pouze čeština. Dalším důležitým adresářem je „*models*“, která obsahuje soubory zpracovávající data, se kterými aplikace pracuje. Jsou zde také třídy implementující vlastní ORM systém. Soubory umístěny v předchozích složkách byly společné jak pro frontend, tak pro administrační část. Složka s názvem „*modules*“ se dále dělí na složky „*admin*“ a „*frontend*“. Jsou zde umístěny třídy controllerů, které se starají o reakce na události od uživatele a také na změnu dat zobrazených ve view. Dále zde najdeme adresář „*layouts*“ obsahující HTML šablony definující rozložení celkové stránky. V adresáři „*views*“ jsou pak šablony pro konkrétní části aplikace.

Knihovny Zend Frameworku a další používané knihovny jsou uloženy v adresáři „library“.

### 5.1.1. Komunikace DroidDrawing s webem

Veškerá komunikace mezi aplikací v mobilním telefonu a webovou prezentací probíhá prostřednictvím jediného controlleru nazvaného „*UzivatelAndroidController*“, který obsahuje akce obsluhující požadavky zasílané mobilním telefonem. Akce, která se má provést, je určena podle tvaru URL adresy dokumentu. Zadáním adresy „*uzivatel-android/upload*“ se například provádí uložení předlohy na Internet. Metodou POST se předají informace o ukládané předloze včetně samotného obrázku uloženého ve formátu PNG. Textové informace jsou uloženy do databáze, obrázek je ukládán ve třech rozměrech přímo do adresářové struktury webového serveru. Do této struktury se uloží originální obrázek, jeho miniatura v rozlišení 180x140px, která je zobrazena v seznamu předloh na webové stránce a nakonec miniatura v rozměrech 67x100px. Tento poslední obrázek je načítán aplikací DroidDrawing v případě, že je zobrazen seznam předloh uložených na Internetu.

K popisu struktury aplikace je to vše. Nyní popíši jednotlivé funkce webové prezentace.

## 5.2. Omalovánky

Tato sekce se zobrazí ihned na úvodní stránce. Je zde seznam předloh, které uživatelé uložili na web (viz *Obrázek 20*).



Obrázek 20 - seznam předloh

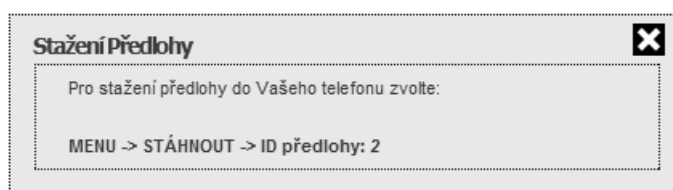


Nad tímto seznamem je umístěn blok „řadit omalovánky od“ (viz *Obrázek 21*) obsahující čtyři možnosti jak předlohy řadit. Můžeme si vybrat řazení od nejstarších, nejnovějších, nejlépe hodnocených a nejhůře hodnocených předloh.

**řadit omalovánky od:** nejnovějších nejstarších nejlépe hodnocených nejhůře hodnocených

Obrázek 21 - řazení předloh

Okno každé předlohy obsahuje její miniaturu s ikonou lupy, která zobrazí předlohu v její originální velikosti. Pod touto miniaturou je název předlohy. Dále jsou zde dva odkazy a to „Stáhnout“ a „Komentáře“. Po kliknutí na „Stáhnout“ se zobrazí dialogové okno (viz *Obrázek 22*) s informací jak stáhnout tuto předlohu do mobilního telefonu.



Obrázek 22 - stažení předlohy

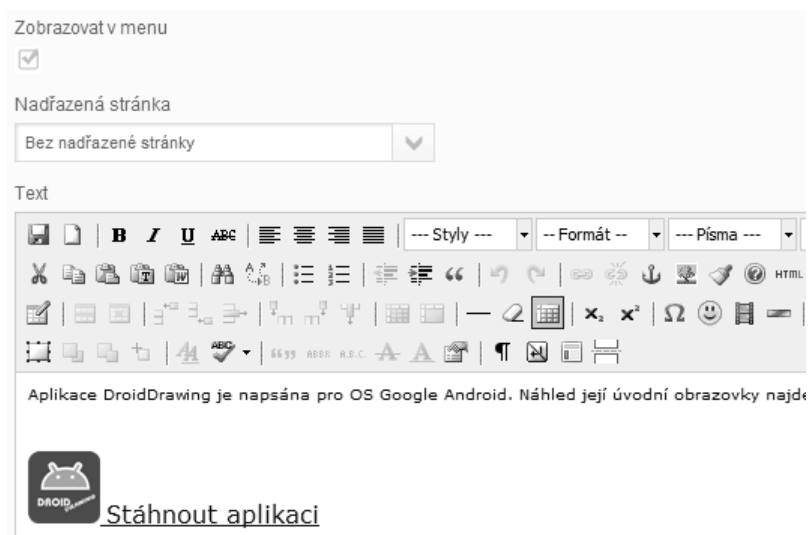
Uživatelé mají možnost uložené předlohy okomentovat, k tomu jim slouží odkaz „Komentáře“. Obsah dialogového okna (viz *Obrázek 23*), které se otevře, je načítán až po kliknutí na odkaz díky technologii AJAX.



Obrázek 23 - komentáře

### 5.3. Ostatní části webu

Dalšími částmi na webu jsou spíše informační stránky obsahující údaje o bakalářské práci a také instalační soubor aplikace ke stažení. Tyto stránky lze snadno upravovat pomocí administračního rozhraní, které obsahuje WYSIWYG editor TinyMCE<sup>15</sup> (viz *Obrázek 24*).



Obrázek 24 - TinyMCE

---

<sup>15</sup> <http://tinymce.moxiecode.com/>

## 6. Závěr

Bakalářská práce se především věnuje vývoji aplikace pro mobilní platformu Google Android OS. Použitím nejnovějších technologií určených pro vývoj aplikace pro tento operační systém jsem vytvořil aplikaci, která by měla odpovídat specifikovaným požadavkům. Snažil jsem se o to, aby ovládání aplikace bylo intuitivní, a také jsem se zaměřil na optimalizaci výkonu. K vytvoření webové prezentace jsem použil vlastní redakční systém, který je napsán v PHP a využívá knihovny Zend Frameworku.

V dalším vývoji aplikace mám v plánu pokračovat. Chtěl bych například zpřístupnit možnost vložit si na pozadí předlohy vlastní obrázky. Časem se chci pokusit aplikaci umístit do obchodu s aplikacemi určených pro Android OS s názvem „*Android Market*“<sup>16</sup>.

Téma této bakalářské práce pro mne bylo velmi zajímavé. Seznámil jsem se s řadou technik používaných při vývoji aplikací nejen pro Android OS, ale také pro mobilní zařízení obecně.

---

<sup>16</sup> <http://www.android.com/market/>

## 7.Literatura

GUTMANS Andi, BAKKEN Stig Seather, RETHANS Derick. *PHP 5 Power Programming*. 1. vyd. Brno: PC Books, 2005.

ULLMAN Larry. *PHP And MySQL For Dynamic Web Sites: Visual Quickpro Guide*. 1. vyd. Brno: Computer Press, 2004.

MEIER Reto. *Professional Android Application Development*. 1. vyd. Indianapolis: Wiley Publishing, 2009.

Google. *What is Android?* [online]. 2009, 12.4.2010 [citováno 23.4.2010].  
<http://developer.android.com/guide/basics/what-is-android.html>.

Přispěvatelé Wikipedie. *Windows Mobile* [online]. Wikipedie, Otevřená encyklopedie; 2010, 23.4.2010 [citováno 23.4.2010].  
[http://en.wikipedia.org/w/index.php?title=Windows\\_Mobile&oldid=357786100](http://en.wikipedia.org/w/index.php?title=Windows_Mobile&oldid=357786100).

Přispěvatelé Wikipedie. *iPhone OS* [online]. Wikipedie, Otevřená encyklopedie; 2010, 22.4.2010 [citováno 23.4.2010].  
[http://en.wikipedia.org/w/index.php?title=IPhone\\_OS&oldid=357530491](http://en.wikipedia.org/w/index.php?title=IPhone_OS&oldid=357530491).

Přispěvatelé Wikipedie. *Zend Framework* [online]. Wikipedie, Otevřená encyklopedie; 2010, 21.4.2010 [citováno 23.4.2010].  
[http://en.wikipedia.org/w/index.php?title=Zend\\_Framework&oldid=357427390](http://en.wikipedia.org/w/index.php?title=Zend_Framework&oldid=357427390).

Open Source Initiative. *The Open Source Definition* [online]. 2009, 25.3.2010 [citováno 23.4.2010]. <http://www.opensource.org/docs/osd>.

Google. *Open Handset Alliance - FAQ* [online]. 2007, listopad 2007 [citováno 23.4.2010].  
[http://www.openhandsetalliance.com/oha\\_faq.html](http://www.openhandsetalliance.com/oha_faq.html).

*Flood Fill* [online]. 2010, 2010 [citováno 23.4.2010].  
<http://www.academictutorials.com/graphics/graphics-flood-fill.asp>.

## 8. Přílohy

### 8.1. PŘÍLOHA A: Hlavní stránka webové prezentace

